The embodiments of the invention in which an exclusive property or privilege is claimed are defined as follows:

1. In a computer system including a display, a user input facility, and an application for presenting user interfaces on the display, a Web-style application comprising:

one or more page functions, each page function comprising:

a set of exposed attributes, wherein first subset of the set of exposed attributes defines types of information receivable by the page function, and a second subset of the set of exposed attributes defines types of information returnable by the page function, and wherein the types of information returnable by the page function are strongly typed;

a set of page function services, including an activate service and a complete service, both services being invocable to execute a decision selected from a group consisting of a decision to finish, a decision to show the user interface page, and a decision to create a new page function; and

user interface elements to be displayed on the display; and

a frame, the frame comprising:

a set of frame services, the set of frame services including a navigate service and a finish service, wherein the navigate service being invocable by a page function to cause the frame to invoke the activate service of another page function and cause the other page function to perform a task, and wherein the finish service being invocable by the other page function to cause the frame to invoke the complete service of the page function; and

a data structure, wherein the data structure stores data that identifies each page function to which the frame has navigated, and relationships among page functions.

2. The Web-style application of Claim 1, wherein the set of exposed attributes includes an identifier that uniquely identifies the page function.

3. The Web-style application of Claim 2, wherein the data structure includes one or more nodes, each node containing the identifier of a corresponding page function to which the frame has navigated, each node further containing a link to another node if the other node cause the frame to navigate to the node.

4. The Web-style application of Claim 1, wherein the frame further comprises a binding to a frame user interface page.

5. The Web-style application of Claim 4, wherein the frame user interface page has a frame window with a periphery and a region within the periphery of the frame window for showing the user interface elements of a page function.

6. The Web-style application of Claim 5, wherein the frame window includes a back facility that, in response to an input received from the user input facility, causes the previously displayed user interface elements of a page function to be redisplayed.

7. In a computer system including a display, a user input facility, and an application for presenting user interfaces on the display, one or more page functions being stored on a computer-readable medium as a data type, each page function comprising:

a set of exposed attributes accessible externally to the page function, wherein a first subset of the set of exposed attributes define types of information receivable by the page function, and a second subset of the set of exposed attributes define types of information returnable by the page function, and wherein the exposed attributes that define types of information returnable by the page function are strongly typed;

a set of page function services including an activate service and a complete service, both services being invocable to execute a decision selected from a group consisting of a decision to finish, a decision to show user interface elements, and a decision to create a new page function; and

user interface elements operable to be displayed on the display.

8. The page function of Claim 7, wherein the activate service has a set of parameters, each corresponding to one of the first subset of exposed attributes.

9. The page function of Claim 7, wherein the complete service has a set of parameters, a portion of the set of parameters identifying another page function created by the page function to perform a second task, another portion of the set of parameters corresponding to one of the second subset of exposed attributes.

10. The page function of Claim 7, further comprising an identifier for identifying each instance of the page function.

11. The page function of Claim 7, wherein the user interface elements may be selectively displayed on the display.

12. In a computer system including a display, a user input facility, and an application for presenting user interfaces on the display, one or more page functions being stored on a computer-readable medium as a data type, a programming system having a computer-readable medium that has stored thereon an architectural software framework, the architectural software framework comprising:

a first data type defined as a page function, the page function comprising:

a set of exposed attributes accessible externally to the page function, wherein a first subset of the set of exposed attributes define types of information receivable by the page function, and a second subset of the set of exposed attributes define types of information returnable by the page function, and wherein the set of exposed attributes are strongly typed;

a set of page function services, wherein the set of page function services including an activate service and a complete service, both services being invocable to execute a decision selected from a group consisting of a decision to finish, a decision to show a user interface page, and a decision to create a new page function; and

user interface elements to be selectively displayed on the display; and

a second data type defined as a frame, the frame comprising

a set of frame services, wherein the set of frame services includes a navigate service and a finish service, the navigate service being invocable by a first page function to navigate to a second page function to perform a task, and the finish service being invocable by the second page function to communicate to the frame that the task has been performed; and

a frame data structure, wherein the frame data structure stores information that identifies each page function to which the frame has navigated, and shows the originator relationship among page functions.

13. A computing environment for displaying user interface elements on an output device in a Web-style manner, the computing environment comprising:

an output device;

an input device; and

a program interface module exposing an interface function that, when invoked:

creates a page function comprising:

a set of exposed attributes, wherein a first subset of the set of exposed attributes defining types of information receivable by the page function, and a second subset of the set of exposed attributes defining types of information returnable by the page function, the exposed attributes defining types of information returnable by the page function being strongly typed;

a set of page function services, wherein the set of page function services including an activate service and a complete service, both services being invocable to execute a decision selected from a group consisting of a decision to finish, a decision to display a user interface page, and a decision to create a new page function; and

user interface elements that may be selectively displayed on the output device; and

selectively displays the user interface elements of the page function on the output device.

14. The computing environment of Claim 13 wherein the exposed interface function is explicitly defined according to a specific type of information returnable by the page function.

15. The computing environment of Claim 14 wherein the specific type of information returnable by the page function is an integer.

16. The computing environment of Claim 14 wherein the specific type of information returnable by the page function is a character string.

17. The computing environment of Claim 14 wherein the specific type of information returnable by the page function is a Boolean value.

18. The computing environment of Claim 13 wherein the exposed interface function requires a parameter for identifying the specific type of information returnable by the page function.

19. The computing environment of Claim 18 wherein the exposed interface function is a template-style interface function.

20. The computing environment of Claim 18 wherein the required parameter for identifying the specific type of information returnable by the page function is an integer identifier, and wherein the specific type of information returnable by the page function is an integer.

21. The computing environment of Claim 18 wherein the required parameter for identifying the specific type of information returnable by the page function is an character string identifier, and wherein the specific type of information returnable by the page function is an character string.

22. The computing environment of Claim 18 wherein the required parameter for identifying the specific type of information returnable by the page function is an Boolean identifier, and wherein the specific type of information returnable by the page function is Boolean value.

23. The computing environment of Claim 13, wherein the first subset of the set of exposed attributes defining types of information receivable by the page function includes an add return delegate function which, when invoked with a parameter identifying a return delegate routine, enables the page function to return its information.

24. A computer-readable medium having computer-executable instructions which, when executed on a computer system implement an application programming interface module exposing a programming interface, such that when the programming interface is invoked:

creates a page function comprising:

a set of exposed attributes, wherein a first subset of the set of exposed attributes defining types of information receivable by the page function, and a second subset of the set of exposed attributes defining types of information returnable by the page function, the exposed attributes defining types of information returnable by the page function being strongly typed;

a set of page function services, wherein the set of page function services including an activate service and a complete service, both services being invocable to execute a decision selected from a group consisting of a decision to finish, a decision to display a user interface page, and a decision to create a new page function; and

user interface elements that may be selectively displayed on the output device; and

selectively displays the user interface elements of the page function on the output device.

25. The programming interface module of Claim 24, wherein the programming interface is explicitly defined according to a specific type of information returnable by the page function.

26. The programming interface module of Claim 25, wherein the specific type of information returnable by the page function is an integer.

27. The programming interface module of Claim 25, wherein the specific type of information returnable by the page function is a character string.

28. The programming interface module of Claim 25, wherein the specific type of information returnable by the page function is a Boolean value.

29. The programming interface module of Claim 24, wherein the programming interface requires a parameter for identifying the specific type of information returnable by the page function.

30. The programming interface module of Claim 29, wherein the exposed interface function is a template-style interface function.

31. The programming interface module of Claim 29, wherein the required parameter for identifying the specific type of information returnable by the page function is an integer identifier, and wherein the specific type of information returnable by the page function is an integer.

32. The programming interface module of Claim 29, wherein the required parameter for identifying the specific type of information returnable by the page function is a character string identifier, and wherein the specific type of information returnable by the page function is a character string.

33. The programming interface module of Claim 29, wherein the required parameter for identifying the specific type of information returnable by the page function is an Boolean identifier, and wherein the specific type of information returnable by the page function is Boolean value.

34. The programming interface module of Claim 24, wherein the first subset of the set of exposed attributes defining types of information receivable by the page function includes an add return delegate function which, when invoked with a parameter identifying a return delegate routine, enables the page function to return its information.